CAN Bus Interface Chip CH9431

Datasheet

Version: 1.3

https://wch-ic.com

1. Overview

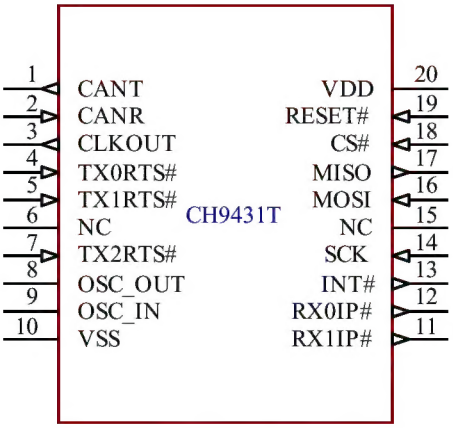
CH9431 is a Controller Area Network (CAN) protocol controller that supports the CAN V2.0B specification; it supports transmitting and receiving standard and extended data frames, as well as remote frames. CH9431 provides 2 receive mask registers and 6 receive filter registers, which filters out unwanted messages and reduces the resource consumption of the main MCU.

CH9431 communicates with the host device through the standard Serial Peripheral Interface (SPI), which can be used to extend the CAN function of the host device to connect to the external CAN bus.

2. Feature

- Support rated operating voltage 3.3V
- Support CAN V2.0B specification, communication rate up to 1Mbps: Support data fields 0 ~ 8 bytes in length, and support transmitting standard, extended and remote frames
- Built-in 2 receive buffers, six 29-bit receive filter registers, and two 29-bit receive mask registers.
- Built-in 3 transmit buffers, all with transmit abort function
- Built-in high-speed SPI interface, frequency support up to 12MHz
- Support single-trigger mode to ensure that messages are transmitted only once
- Support light sleep and deep sleep modes to save power. Shallow sleep and deep sleep modes are supported to save power consumption. Shallow sleep mode will not lose the CAN packets that wake up the controller.
- Industrial temperature range: -40 ~ 85℃
- TSSOP20 package is available

3. Pinouts



Package form	Body size	Pin pitch		Package description	Order model
TSSOP20	4.4*6.5mm	0.65mm	25.6mil	Quad Flat No-lead Package	CH9434A

4. Pin Definitions

Table 4-1 Pin definitions

Pin No.	Pin name	Type	Pin description
1	CANT	O (FT)	CAN bus transmit output pins.
2	CANR	I (FT)	CAN bus receive input pins.
3	CLKOUT	O (FT)	Clock output pin with programmable prescaler.
4	TX0RTS#	I, PU (FT)	Transmit buffer TXB0 request transmit pin or general-purpose digital input pin with built-in pull-up resistor.
5	TX1RTS#	I, PU (FT)	Transmit buffer TXB1 request transmit pin or general-purpose digital input pin with built-in pull-up resistor.
6	NC	NC	Suspend, no connection.
7	TX2RTS#	I, PU (FT)	Transmit buffer TXB2 request transmit pin or general-purpose digital input pin with built-in pull-up resistor.
8	OSC_OUT	O	Crystal inverted output, need to be connected to one end of the external 16MHz crystal, the load capacitance of the crystal is based on the crystal manufacturer's recommendations.
9	OSC_IN	I	Crystal input, need to be externally connected to the other end of the 16MHz crystal, the load capacitance of the crystal is based on the crystal manufacturer's recommendation.
10	VSS	P	Common ground
11	RX1IP#	O	Receive buffer RXB1 interrupt pin or general-purpose digital output pin.

12	RX0IP#	O	Receive buffer RXB0 interrupt pin or general-purpose digital output pin.
13	INT#	O	Interrupt signal output pin.
14	SCK	I	Clock input pin of SPI interface.
15	NC	NC	Suspend, no connection.
16	MOSI	I, PU, PD	Data input pin of SPI interface with built-in pull-up resistor, which automatically switches to the built-in pull-down resistor in sleep mode.
17	MISO	O	Data output pin of SPI interface.
18	CS#	I	Chip selection input pin of SPI interface, active low.
19	RESET#	I, PU	External reset input, active low, built-in pull-up resistor.
20	VDD	P	Rated 3.3V power input.

Note: Pin type abbreviations:

I = Signal input; O = Signal output; P = Power or Ground;

FT = Tolerant 5V; PD = Built-in pull-down resistor; PU = Built-in pull-up resistor.

5. Register

The microcontroller referred to in this manual basically applies to DSP or MCU/MPU/CPU etc.

The CH9431's internal registers as well as buffers are allocated in the address range from 00h to 0FFh and are accessed by the microcontroller after addressing.

The default values after reset are expressed as binary numbers and can be characterized by several character flags, which are as follows:

0 = Bit set to logic 0;

1 = Bit set to logic 1;

RO = Read-only;

RW = Read/write;

Y = Support bit-modify commands.

5.1 Serial Register

Register address	Register name	Register description	Bit modification
00h	R8_RXF0SIDL	Receive filter register 0 standard identifier low bit	
01h	R8_RXF0SIDH	Receive filter register 0 standard identifier high bit	
02h	R8_RXF0EIDL	Receive filter register 0 extended identifier low bit	
03h	R8_RXF0EIDH	Receive filter register 0 extended identifier high bit	
04h	R8_RXF1SIDL	Receive filter register 1 standard identifier low bit	
05h	R8_RXF1SIDH	Receive filter register 1 standard identifier high bit	
06h	R8_RXF1EIDL	Receive filter register 1 extended identifier low bit	
07h	R8_RXF1EIDH	Receive filter register 1 extended identifier high bit	
08h	R8_RXF2SIDL	Receive filter register 2 standard identifier low bit	
09h	R8_RXF2SIDH	Receive filter register 2 standard identifier high bit	
0Ah	R8_RXF2EIDL	Receive filter register 2 extended identifier low bit	
0Bh	R8_RXF2EIDH	Receive filter register 2 extended identifier high bit	
0Ch	R8_RXIPCTRL	RXnBF pin control register and status register	Y
0Dh	R8_TXRTSCTRL	TXnRTS pin control and status register	Y
0Eh	R8_SYSSTAT	System status register	
0Fh	R8_SYSCTRL	System control register	Y
10h	R8_RXF3SIDL	Receive filter register 3 standard identifier low bit	
11h	R8_RXF3SIDH	Receive filter register 3 standard identifier high bit	
12h	R8_RXF3EIDL	Receive filter register 3 extended identifier low bit	
13h	R8_RXF3EIDH	Receive filter register 3 extended identifier high bit	
14h	R8_RXF4SIDL	Receive filter register 4 standard identifier low bit	
15h	R8_RXF4SIDH	Receive filter register 4 standard identifier high bit	
16h	R8_RXF4EIDL	Receive filter register 4 extended identifier low bit	
17h	R8_RXF4EIDH	Receive filter register 4 extended identifier high bit	
18h	R8_RXF5SIDL	Receive filter register 5 standard identifier low bit	
19h	R8_RXF5SIDH	Receive filter register 5 standard identifier high bit	
1Ah	R8_RXF5EIDL	Receive filter register 5 extended identifier low bit	

1Bh	R8_RXF5EIDH	Receive filter register 5 extended identifier high bit	
1Ch	R8_TEC	Transmit error counter	
1Dh	R8_REC	Receive error counter	
1Eh	R8_SYSSTAT	System status register	
1Fh	R8_SYSCTRL	System control register	Y
20h	R8_RXM0SIDL	Receive mask register 0 standard identifier low bit	
21h	R8_RXM0SIDH	Receive mask register 0 standard identifier high bit	
22h	R8_RXM0EIDL	Receive mask register 0 extended identifier low bit	
23h	R8_RXM0EIDH	Receive mask register 0 extended identifier high bit	
24h	R8_RXM1SIDL	Receive mask register 1 standard identifier low bit	
25h	R8_RXM1SIDH	Receive mask register 1 standard identifier high bit	
26h	R8_RXM1EIDL	Receive mask register 1 extended identifier low bit	
27h	R8_RXM1EIDH	Receive mask register 1 extended identifier high bit	
28h	R8_BTIMER3	Configuration register 3	Y
29h	R8_BTIMER2	Configuration register 2	Y
2Ah	R8_BTIMER1	Configuration register 1	Y
2Bh	R8_SYSINTE	Interrupt enable register	Y
2Ch	R8_SYSINTF	Interrupt flag register	Y
2Dh	R8_EFLAG	Error flag register	Y
2Eh	R8_SYSSTAT	System status register	
2Fh	R8_SYSCTRL	System control register	Y
30h	R8_TXB0CTRL	Transmit buffer 0 control register	Y
31h	R8_TXB0SIDL	Transmit buffer 0 standard identifier low bit	
32h	R8_TXB0SIDH	Transmit buffer 0 standard identifier high bit	
33h	R8_TXB0EIDL	Transmit buffer 0 extended identifier low bit	
34h	R8_TXB0EIDH	Transmit buffer 0 extended identifier high bit	
35h	R8_TXB0DLC	Transmit buffer 0 length register	
36h	R8_TXB0D0	Transmit buffer 0 data byte 0	
37h	R8_TXB0D1	Transmit buffer 0 data byte 1	
38h	R8_TXB0D2	Transmit buffer 0 data byte 2	
39h	R8_TXB0D3	Transmit buffer 0 data byte 3	
3Ah	R8_TXB0D4	Transmit buffer 0 data byte 4	
3Bh	R8_TXB0D5	Transmit buffer 0 data byte 5	
3Ch	R8_TXB0D6	Transmit buffer 0 data byte 6	
3Dh	R8_TXB0D7	Transmit buffer 0 data byte 7	
3Eh	R8_SYSSTAT	System status register	
3Fh	R8_SYSCTRL	System control register	Y
40h	R8_TXB1CTRL	Transmit buffer 1 control register	Y
41h	R8_TXB1SIDL	Transmit buffer 1 standard identifier low bit	
42h	R8_TXB1SIDH	Transmit buffer 1 standard identifier high bit	
43h	R8_TXB1EIDL	Transmit buffer 1 extended identifier low bit	
44h	R8_TXB1EIDH	Transmit buffer 1 extended identifier high bit	
45h	R8_TXB1DLC	Transmit buffer 1 length register	

46h	R8_TXB1D0	Transmit buffer 1 data byte 0	
47h	R8_TXB1D1	Transmit buffer 1 data byte 1	
48h	R8_TXB1D2	Transmit buffer 1 data byte 2	
49h	R8_TXB1D3	Transmit buffer 1 data byte 3	
4Ah	R8_TXB1D4	Transmit buffer 1 data byte 4	
4Bh	R8_TXB1D5	Transmit buffer 1 data byte 5	
4Ch	R8_TXB1D6	Transmit buffer 1 data byte 6	
4Dh	R8_TXB1D7	Transmit buffer 1 data byte 7	
4Eh	R8_SYSSTAT	System status register	
4Fh	R8_SYSCTRL	System control register	Y
50h	R8_TXB2CTRL	Transmit buffer 2 control register	Y
51h	R8_TXB2SIDL	Transmit buffer 2 standard identifier low bit	
52h	R8_TXB2SIDH	Transmit buffer 2 standard identifier high bit	
53h	R8_TXB2EIDL	Transmit buffer 2 extended identifier low bit	
54h	R8_TXB2EIDH	Transmit buffer 2 extended identifier high bit	
55h	R8_TXB2DLC	Transmit buffer 2 length register	
56h	R8_TXB2D0	Transmit buffer 2 data byte 0	
57h	R8_TXB2D1	Transmit buffer 2 data byte 1	
58h	R8_TXB2D2	Transmit buffer 2 data byte 2	
59h	R8_TXB2D3	Transmit buffer 2 data byte 3	
5Ah	R8_TXB2D4	Transmit buffer 2 data byte 4	
5Bh	R8_TXB2D5	Transmit buffer 2 data byte 5	
5Ch	R8_TXB2D6	Transmit buffer 2 data byte 6	
5Dh	R8_TXB2D7	Transmit buffer 2 data byte 7	
5Eh	R8_SYSSTAT	System status register	
5Fh	R8_SYSCTRL	System control register	Y
60h	R8_RXB0CTRL	Receive buffer 0 control register	Y
61h	R8_RXB0SIDL	Receive buffer 0 standard identifier low bit	
62h	R8_RXB0SIDH	Receive buffer 0 standard identifier high bit	
63h	R8_RXB0EIDL	Receive buffer 0 extended identifier low bit	
64h	R8_RXB0EIDH	Receive buffer 0 extended identifier high bit	
65h	R8_RXB0DLC	Receive buffer 0 length register	
66h	R8_RXB0D0	Receive buffer 0 data byte 0	
67h	R8_RXB0D1	Receive buffer 0 data byte 1	
68h	R8_RXB0D2	Receive buffer 0 data byte 2	
69h	R8_RXB0D3	Receive buffer 0 data byte 3	
6Ah	R8_RXB0D4	Receive buffer 0 data byte 4	
6Bh	R8_RXB0D5	Receive buffer 0 data byte 5	
6Ch	R8_RXB0D6	Receive buffer 0 data byte 6	
6Dh	R8_RXB0D7	Receive buffer 0 data byte 7	
6Eh	R8_SYSSTAT	System status register	
6Fh	R8_SYSCTRL	System control register	Y
70h	R8_RXB1CTRL	Receive buffer 1 control register	

71h	R8_RXB1SIDL	Receive buffer 1 standard identifier low bit	
72h	R8_RXB1SIDH	Receive buffer 1 standard identifier high bit	
73h	R8_RXB1EIDL	Receive buffer 1 extended identifier low bit	
74h	R8_RXB1EIDH	Receive buffer 1 extended identifier high bit	
75h	R8_RXB1DLC	Receive buffer 1 length register	
76h	R8_RXB1D0	Receive buffer 1 data byte 0	
77h	R8_RXB1D1	Receive buffer 1 data byte 1	
78h	R8_RXB1D2	Receive buffer 1 data byte 2	
79h	R8_RXB1D3	Receive buffer 1 data byte 3	
7Ah	R8_RXB1D4	Receive buffer 1 data byte 4	
7Bh	R8_RXB1D5	Receive buffer 1 data byte 5	
7Ch	R8_RXB1D6	Receive buffer 1 data byte 6	
7Dh	R8_RXB1D7	Receive buffer 1 data byte 7	
7Eh	R8_SYSSTAT	System status register	
7Fh	R8_SYSCTRL	System control register	Y

5.2 System Control Register (R8_SYSCTRL)

Bit	Name	Description	Access	Default value
[7:5]	REQOP	Request Operating Mode Selection: 000: Set to normal operating mode; 001: Set to configuration mode; 010: Set to listen only mode; 011: Set to loopback mode; 100: Set to shallow hibernation mode; 101: Set to deep hibernation mode; Other values are invalid.	RW	001b
4	ABAT	Abort all current message transmissions bit: 1: Request to abort all current message transmissions in the buffer; 0: Terminate the request to abort all message transmissions.	RW	0
3	OSM	Single Trigger Mode Enable Bit: 1: Enable. Messages are attempted to be sent only once; 0: Disable. If necessary, the message will be resent.	RW	0
2	CLKEN	CLKOUT pin enable bit: 1: CLKOUT pin enabled; 0: CLKOUT pin disabled (pin in high impedance state).	RW	1
[1:0]	CLKPRE	CLKOUT pin prescaler ratio setting: 00: FCLKOUT = 20MHz; 01: FCLKOUT = 10MHz; 10: FCLKOUT = 8MHz; 11: FCLKOUT = 4MHz。	RW	11b

5.3 System Status Register (R8_SYSSTAT)

Bit	Name	Description	Access	Default value
[7:5]	OPMOD	Operating Mode Bit Selection: 000: The device is in normal operating mode; 001: The device is in configuration mode; 010: The device is in listen-only mode; 011: The device is in loopback mode; 110: The device is in firmware update mode; The SPI cannot be read when it is asleep, so the other values are invalid.	RO	001b
4	Reserved	Reserved	RO	0
[3:1]	ICOD	Interrupt flag code bits: 000: No interrupt; 001: Error interrupt; 010: Wake-up interrupt; 011: TXB0 interrupt; 100: TXB1 interrupt; 101: TXB2 interrupt; 110: RXB0 interrupt; 111: RXB1 interrupt.	RO	0
0	Reserved	Reserved	RO	0

5.4 RXnIP Pin Control Register and Status Register (R8_RXIPCTRL)

Bit	Name	Description	Access	Default value
[7:6]	Reserved	Reserved	RO	0
5	B1BFS	RX1BF pin status bit (digital output operating mode only): 1: Output high level; 0: Output low level.	RW	0
4	B0BFS	RX0BF pin status bits (digital output operation mode only): 1: Output high level; 0: Output low level.	RW	0
3	B1BFE	RX1BF pin function enable bit: 1: Pin function enabled, the operation mode is determined by the B1BFM bit; 0: Pin function disabled, the pin is in high resistance state.	RW	0
2	B0BFE	RX0BF pin function enable: 1: Pin function enabled, the operation mode is determined by the B0BFM bit; 0: Pin function disabled, the pin is in high resistance state.	RW	0
1	B1BFM	RX1BF pin operating modes: 1: This pin is used to generate an interrupt when a valid message is loaded into RXB1;	RW	0

		0: Digital output mode.		
0	B0BFM	RX0BF pin operation modes: 1: This pin is used to generate an interrupt when a valid message is loaded into RXB0; 0: Digital output mode.	RW	0

5.5 Configuration Register 1 (R8_BTIMER1)

Bit	Name	Description	Access	Default value
[7:0]	BRP	Minimum time cell length setting value: Tq = (BRP+1) × (1/20Mhz) *Note: The CAN baud rate is calculated by the formula:* *CANbps = 20Mhz / ((TS1+1+TS2+1+1)*(BRP+1))*	RW	0

5.6 Configuration Register 2 (R8_BTIMER2)

Bit	Name	Description	Access	Default value
[7:6]	Reserved	Reserved	RO	0
[5:4]	SJW	Re-synchronize jump widths: 11: Width = 4 × TQ; 10: Width = 3 × TQ; 01: Width = 2 × TQ; 00: Width = 1 × TQ.	RW	0
[3:0]	TS1	Propagation time period and phase buffer segment 1-bit time duration: (TS1 + 1) × TQ	RW	0

5.7 Configuration Register 3 (R8_BTIMER3)

Bit	Name	Description	Access	Default value
[7:3]	Reserved	Reserved	RO	0
[2:0]	TS2	Phase buffer segment 2 length: (TS2 + 1) × TQ	RW	0

5.8 Interrupt Enable Register (R8_SYSINTE)

Bit	Name	Description	Access	Default value
7	MERRE	Message error interrupt enable bit: 1: Interrupt when an error occurs during message reception or transmission; 0: Disable.	RW	0
6	WAKIE	Wake-up interrupt enable bit: 1: Interrupt when there is activity on the CAN bus;	RW	0

		0: Disabled.		
5	ERRIE	Error interrupt enable bit (multiple interrupt sources in the EFLAG register): 1: Interrupt on change of EFLAG error condition; 0: Disabled.	RW	0
4	TX2IE	Transmit buffer 2 empty interrupt enable bit: 1: Interrupt when TXB2 is empty; 0: Disable.	RW	0
3	TX1IE	Transmit Buffer 1 empty interrupt enable bit: 1 = Interrupt when TXB1 is empty; 0 = Disable.	RW	0
2	TX0IE	Transmit Buffer 0 empty interrupt enable bit: 1 = Interrupt when TXB0 is empty; 0 = Disable.	RW	0
1	RX1IE	Receive buffer 1 full interrupt enable bit: 1: Interrupt when RXB1 loads a message; 0: Disable.	RW	0
0	RX0IE	Receive buffer 0 full interrupt enable bit: 1: Interrupt when RXB0 loads a message; 0: Disable.	RW	0

5.9 Interrupt Flag Register (R8_SYSINTF)

Bit	Name	Description	Access	Default value
7	MERRF	Message Error Interrupt Flag Bit: 1: Interrupt with pending processing (this bit is set by hardware and must be cleared by the MCU to reset the interrupt); 0: Interrupt with no pending processing.	RW	0
6	WAKIF	Wake-up interrupt flag bits: 1: Interrupt with pending processing (this bit is set by hardware and must be cleared by the MCU to reset the interrupt); 0: Interrupt with no pending processing.	RW	0
5	ERRIF	Error interrupt flag bits (multiple interrupt sources in EFLAG register): 1: Interrupt with pending processing (this bit is set by hardware and must be cleared by the MCU to reset the interrupt); 0: Interrupt with no pending processing.	RW	0
4	TX2IF	Transmit buffer 2 Air interrupt flag bit: 1: Interrupt with pending processing (this bit is set by hardware and must be cleared by the MCU to reset the interrupt);	RW	0

		0: Interrupt with no pending processing.		
3	TX1IF	Transmit buffer 1 air interrupt flag bit: 1: There is an interrupt waiting to be processed (this bit is set by the hardware and must be cleared by the MCU to reset the interrupt); 0: There is no interrupt waiting to be processed.	RW	0
2	TX0IF	Transmit buffer 0 air interrupt flag bit: 1: There is an interrupt waiting to be processed (this bit is set by the hardware and must be cleared by the MCU to reset the interrupt); 0: There is no interrupt waiting to be processed.	RW	0
1	RX1IF	Receive buffer 1 air interrupt flag bit: 1: There is an interrupt waiting to be processed (this bit is set by the hardware and must be cleared by the MCU to reset the interrupt); 0: There is no interrupt waiting to be processed.	RW	0
0	RX0IF	Receive buffer 0 air interrupt flag bit: 1: There is an interrupt waiting to be processed (this bit is set by the hardware and must be cleared by the MCU to reset the interrupt); 0: There is no interrupt waiting to be processed.	RW	0

5.10 Error Flag Register (R8_EFLAG)

Bit	Name	Description	Access	Default value
7	RX1OVR	Receive buffer 1 Overflow Flag Bit: 1: RXB1 receives a valid message and the RX1IF bit of R8_SYSINTF is 1; 0: Must be reset by the MCU.	RW	0
6	RX0OVR	Receive buffer 0 Overflow Flag Bit: 1: RXB0 receives a valid message and the RX0IF bit of R8_SYSINTF is 1; 0: Must be reset by the MCU.	RW	0
5	TXBO	Bus shutdown error flag bit: 1: This bit 1 when TEC reaches 255; 0: Reset when a bus recovery sequence is successfully received.	RW	0
4	TXEP	Transmit passive error flag bit: 1: TEC ≥ 128; 0: TEC < 128.	RW	0
3	RXEP	Receive passive error flag bit: 1: REC ≥ 128; 0: TEC < 128.	RW	0
2	TXWAR	Transmit error warning flag bit:	RW	0

		1: TEC ≥ 96; 0: TEC < 96.		
1	RXWAR	Receive error warning flag bit: 1: REC ≥ 96; 0: REC < 96.	RW	0
0	EWARN	Error warning flag bit: 1: TEC and REC ≥ 96; 0: TEC and REC < 96.	RW	0

5.11 TXnRTS Pin Control and Status Register (R8_TXRTSCTRL)

Bit	Name	Description	Access	Default value
[7:6]	FIT	0 = 10ms filter; 1 = 5ms filter; 2 = 1ms filter;; 3 = Do not enable the filter.	RW	0
5	B2RTS	TX2RTS pin status bit: When TX2RTS is in digital input mode, the readout value is the filtered valid value of the pin (low level is 1).	RW	0
4	B1RTS	TX1RTS pin status bit: When TX1RTS is in digital input mode, the readout value is the filtered valid value of the pin (low level is 1).	RW	0
3	B0RTS	TX0RTS pin status bit: When TX0RTS is in digital input mode, the readout value is the filtered valid value of this pin (low level is 1).	RW	0
2	B2RTSM	TX2RTS pin mode bit: 1= This pin is used to request the TXB2 buffer to send a message (on the falling edge after filtering); 0= Digital input.	RW	0
1	B1RTSM	TX1RTS pin mode bit: 1= This pin is used to request the TXB1 buffer to send a message (on the falling edge after filtering); 0= Digital input.	RW	0
0	B0RTSM	TX0RTS pin mode bit: 1= This pin is used to request the TXB0 buffer to send a message (on the falling edge after filtering); 0= Digital input.	RW	0

5.12 Low Standard Identifier of Receiving Filter Register X (R8_RXFxSIDL) (x = 0,1,2,3,4,5)

Bit	Name	Description	Access	Default value
[7:0]	SID[7:0]	Standard identifier filter bit [7:0]:	RW	0

		Data bits that filter the standard identifier data bits [7:0] in the received message.		

5.13 High Standard Identifier of Receiving Filter Register X (R8_RXFxSIDH) (x = 0,1,2,3,4,5)

Bit	Name	Description	Access	Default value
[7:6]	EID[1:0]	Extended identifier filter bit [1:0]: Data bits for filtering extended identifier data bits [1:0] in the received message.	RW	0
5	Reserved	Reserved	RO	0
4	EXIDE	Extended identifier enable bits: 1: Message filtering is only applied to extended frames; 0: Message filtering is only applied to standard frames.	RW	0
3	EN	Whether to enable this filter group: 1: Enabled; 0: Not enabled.	RW	0
[2:0]	SID[10:8]	Standard identifier filter bit [10:8]: Data bits that filter the standard identifier data bits [10:8] in the received message.	RW	0

5.14 Low Extended Identifier of Receiving Filter Register X (R8_RXFxEIDL) (x = 0,1,2,3,4,5)

Bit	Name	Description	Access	Default value
[7:0]	EID[9:2]	Extended identifier filter bit [9:2]: Data bits that filter the standard identifier data bits [9:2] in the received message.	RW	0

5.15 High Extended Identifier of Receiving Filter Register X (R8_RXFxEIDH) (x = 0,1,2,3,4,5)

Bit	Name	Description	Access	Default value
[7:0]	EID[17:10]	Extended identifier filter bit [17:10]: Data bits that filter the standard identifier data bits [17:10] in the received message.	RW	0

5.16 Low Standard Identifier of Receiving Mask Register X (R8_RXMxSIDL) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	SID[7:0]	Standard identifier receive filtered mask bits [7:0]: Data bits that filter-mask the standard identifier data bits [7:0] in the received message.	RW	0

5.17 High Standard Identifier of Receiving Mask Register X (R8_RXMxSIDH) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:6]	EID[1:0]	Extended identifier receive filtered mask bits [1:0]: Data bits for filter masking of the extended identifier data bits [1:0] in the receive message.	RW	0
[5:3]	Reserved	Reserved	RO	0
[2:0]	SID[10:8]	Standard identifier receive filtered mask bits [10:8]: Data bits that filter-mask the standard identifier data bits [10:8] in the received message.	RW	0

5.18 Low Extended Identifier of Receiving Mask Register X (R8_RXMxEIDL) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	EID[9:2]	Extended identifier receive mask bit [9:2]: Data bits for filter masking of the extended identifier data bits [9:2] in the receive message.	RW	0

5.19 High Extended Identifier of Receiving Mask Register X (R8_RXMxEIDH) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	EID[17:10]	Extended identifier receive mask bit [17:10]: Data bits for filter masking of the extended identifier data bits [17:10] in the receive message.	RW	0

5.20 Transmit Buffer x Control Register (R8_TXBxCTRL) (x = 0,1,2)

Bit	Name	Description	Access	Default value

[7:3]	Reserved	Reserved	RO	0
2	MLOA	Message arbitration failure bit: 1: Arbitration failed during message transmission; 0: Arbitration did not fail during message transmission.	RO	0
1	TXERR	Transmit error bit detected: 1: A bus error occurred during message transmission; 0: No bus error occurred during message transmission.	RO	0
0	TXREQ	Message transmit request bit: 1: Buffer waiting for message to be sent (MCU sets this bit to 1 to request message to be transmitted - this bit is cleared automatically after message is transmitted); 0: Buffer not waiting for message to be transmitted (MCU clears this bit to request aborting message transmitting).	RW	0

5.21 Low Standard Identifier of Transmitting Buffer x (R8_TXBxSIDL) (x = 0,1,2)

Bit	Name	Description	Access	Default value
[7:0]	SID[7:0]	Standard identifier data bits of the transmit buffer [7:0]	RW	0

5.22 High Standard Identifier of Transmitting Buffer x (R8_TXBxSIDH) (x = 0,1,2)

Bit	Name	Description	Access	Default value
[7:6]	EID[1:0]	Extended identifier data bits of transmitting buffer [1:0].	RW	0
5	Reserved	Reserved	RO	0
4	EXIDE	Extended identifier enabling bits: 1: The message will transmit the extended identifier; 0: The message will transmit a standard identifier.	RW	0
3	Reserved	Reserved	RO	0
[2:0]	SID[10:8]	Standard identifier data bits of transmitting buffer [10:8]。	RW	0

5.23 Low Extended Identifier of Transmitting Buffer x (R8_TXBxEIDL) (x = 0,1,2)

Bit	Name	Description	Access	Default value
[7:0]	EID[9:2]	Extended identifier data bits of transmitting buffer [9:2].	RW	0

5.24 High Extended Identifier of Transmitting Buffer x (R8_TXBxEIDH) (x = 0,1,2)

Bit	Name	Description	Access	Default value
[7:0]	EID[17:10]	Extended identifier data bits of transmitting buffer [17:10].	RW	0

5.25 Transmit Buffer x Length Register (R8_TXBxDLC) (x = 0,1,2)

Bit	Name	Description	Access	Default value
7	Reserved	Reserved	RO	0
6	RTR	Remote transmitting request bits: 1: The message transmitted is a remote frame; 0: The message transmitted is a data frame.	RW	0
[5:4]	Reserved	Reserved	RO	0
[3:0]	DLC	Data length code: Set the data length to be transmitted (0 to 8 bytes). *Note: DLC can be set to a value greater than 8, but only 8 bytes are sent.*	RW	0

5.26 Receive Buffer 0 Control Register (R8_RXB0CTRL)

Bit	Name	Description	Access	Default value
[7:3]	Reserved	Reserved	RO	0
2	BUKT	Rollover enable bit: 1: If RXB0 is full, the message received by RXB0 will be scrolled to RXB1; 0: Rollover is prohibited.	RW	0
1	Reserved	Reserved	RO	0
0	FILHIT	Filter hit bit, indicating the number of receiving filter register that enables message reception: 1: Receive filter register 1 (RXF1); 0: Receive filter register 0(RXF0).	RO	0

5.27 Low Standard Identifier of Receiving Buffer x (R8_RXBxSIDL) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	SID[7:0]	Receiving data bits of standard identifier bits in a message [7:0].	RO	0

5.28 High Standard Identifier of Receiving Buffer x (R8_RXBxSIDH) (x = 0,1)

Bit	Name	Description	Access	Default value

[7:6]	EID[1:0]	Extended identifier data bits of receiving register [1:0]。	RO	0
5	Reserved	Reserved	RO	0
4	EXIDE	Flag bits of extended identifier: 1: The received message is an extended frame; 0: The received message is a standard frame.	RO	0
3	Reserved	Reserved	RO	0
[2:0]	SID[10:8]	Standard identifier data bits of receiving register [10:8]。	RO	0

5.29 Extended Identifier of Receiving Buffer x (R8_RXBxEIDL) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	EID[9:2]	Extended identifier data bits of receiving register [9:2].	RO	0

5.30 Extended Identifier of Receiving Buffer x (R8_RXBxEIDH) (x = 0,1)

Bit	Name	Description	Access	Default value
[7:0]	EID[17:10]	Extended identifier data bits of receiving register [17:10].	RO	0

5.31 Receive Buffer x Length Register (R8_RXBxDLC) (x = 0,1)

Bit	Name	Description	Access	Default value
7	Reserved	Reserved	RO	0
6	RTR	Remote transmission request bits: 1: Remote (transmission request) frame is received; 0: The received message is a data frame.	RO	0
[5:4]	Reserved	Reserved	RO	0
[3:0]	DLC	Data length code: Set the data length to be transmitted (0 to 8 bytes). *Note: DLC can be set to a value greater than 8, but only 8 bytes are sent.*	RO	0

5.32 Receive Buffer 1 Control Register (R8_RXB1CTRL)

Bit	Name	Description	Access	Default value
[7:3]	Reserved	Reserved	RO	0
[2:0]	FILHIT	Filter hit bit, indicating the receiving filter register number that enables message reception: 101: Receive filter register 5 (RXF5); 100: Receive filter register 4 (RXF4); 011: Receive filter register 3 (RXF5); 010: Receive filter register 2 (RXF2); 001: Receive filter register 1 (RXF1) (Only valid when the BUKT bit in R8_RXB0CTRL is 1);	RO	0

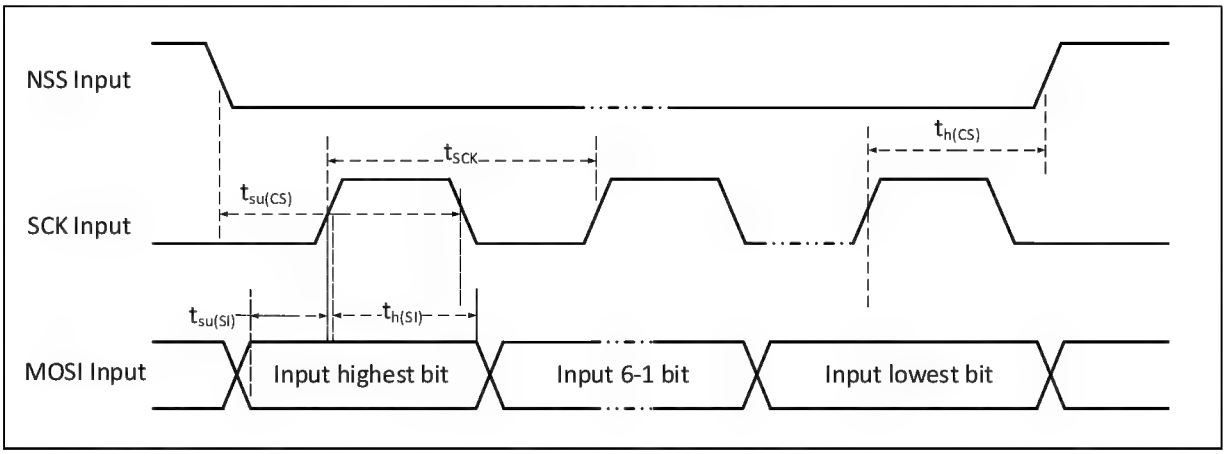
		000: Receive filter register 0 (RXF0) (Only valid when the BUKT bit in R8_RXB0CTRL is 1);		

6. SPI Function Description

6.1 SPI Serial Interface

CH9431 provides a general-purpose SPI synchronous serial interface on the local side, supporting modes 0 and 3. Data is sent with the MSB in front and the LSB behind, external data and commands are transferred into the device via the SI pin, and the data is transferred in on the rising edge of the SCK clock signal. CH9431 is transferred out on the falling edge of SCK via the SO pin. The CS pin must be held low during any SPI operation.

Figure 6-1 SPI timing diagram



When the CS pin is set low, the first byte received by the CH9431 is a command byte, so in successive operations, you must first pull the CS pin high and then wait for the TSX time before pulling the CS pin low to invoke another command. If the command is an illegal command, all data written when the CS pin is low this time is ignored.

After each operation of the CH9431 by a command, at least T_{SX} time shall elapse before the next command operation. Where the RTS command operation shall be followed by an interval of at least T_{DRTS} time.

In the read command, after the address byte is issued, it should wait for at least TSC time before continuing to read the data via SPI.

In the read RX buffer command, after issuing the command byte, it should wait for at least TSRB time before continuing to read the data via SPI.

In the read status instruction and read RX status instruction, wait at least TRS time after issuing the command byte before continuing to read data via SPI.

The interrupt request output from the INT# pin of the CH9431 chip is active low and can be connected to the interrupt input pin or normal input pin of the microcontroller, and the microcontroller can use the interrupt method or the query method to be informed of the interrupt request of the CH9431. In order to save pins, the microcontroller cannot connect to the INT# pin of the CH9431, but directly query the flag registers of the CH9431 such as transmit and receive.

The SI pin is a pull-up input by default (the chip's internal pull-up resistor) and is high when the host reads SI using the float input mode. After entering sleep mode, the SI pin switches to pull-down input and is low when the host reads SI using the float input mode. Therefore, after issuing the sleep command, you can read the current SI

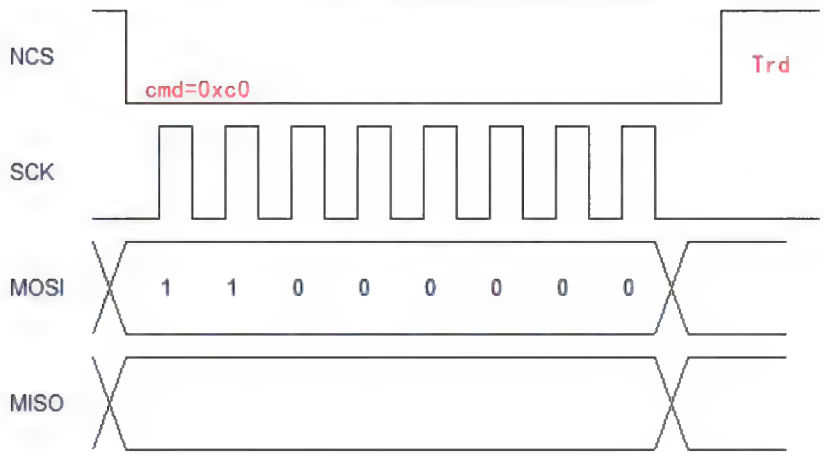
pin level to determine whether you have successfully entered sleep. After exiting sleep, SI switches back to pull-up input mode.

6.2 Reset Instruction

The SPI synchronous reset command re-initializes the CH9431's internal registers and sets it to configuration mode. This command is sent to the CH9431 through the SPI interface and functions in the same way as the RESET pin.

The reset command is a single-byte command. The CS pin should be set low to select the device, then the command byte should be sent, and then the CS pin should be pulled up to a high level after the command has been sent. It is recommended that the reset command be sent (or the RESET pin be held low) as part of the initialization process. After reset, wait for a specified time, T_{rd}, before using the SPI to initialize the device.

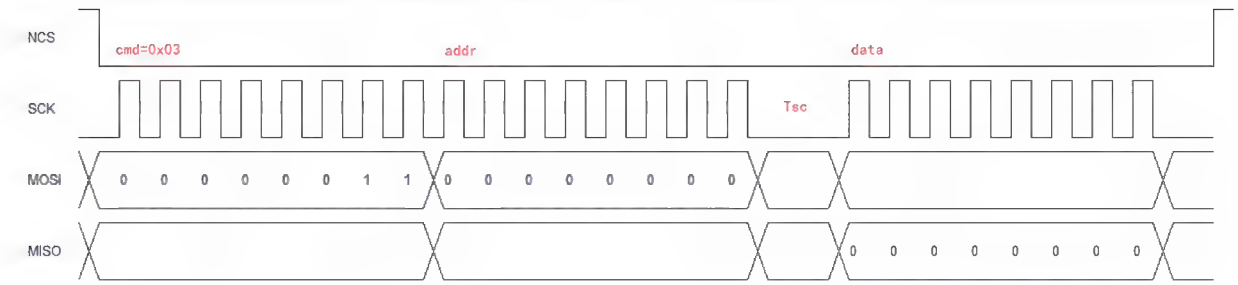
Figure 6-2 Reset instruction timing diagram



6.3 Read Register Instruction

Set the CS pin low to initiate a read instruction. Send the read instruction to CH9431 first, and then continue to send the register address. After sending the address, you need to wait for the specified time Tsc, and then you can read the data at the specified address from the SO pin.

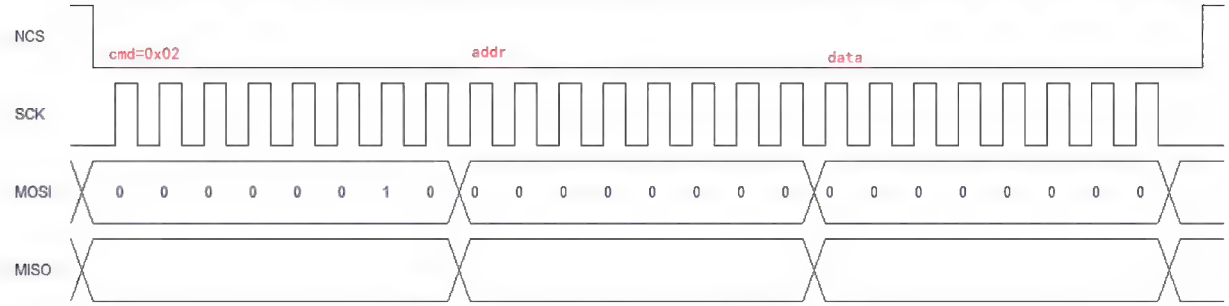
Figure 6-3 Read register instruction timing diagram



6.4 Write Register Instruction

Set the CS pin low to initiate a write operation. Send a write instruction to CH9431 first, then proceed to send the register address and the data to be written to the specified address. After the write operation, the registers are not updated immediately, and it is necessary to wait for the program checksum before updating the written data to the specified registers. Therefore, it is not possible to perform a large number of write or bit modification operations to the register in a short period of time.

Figure 6-4 Write register instruction timing diagram



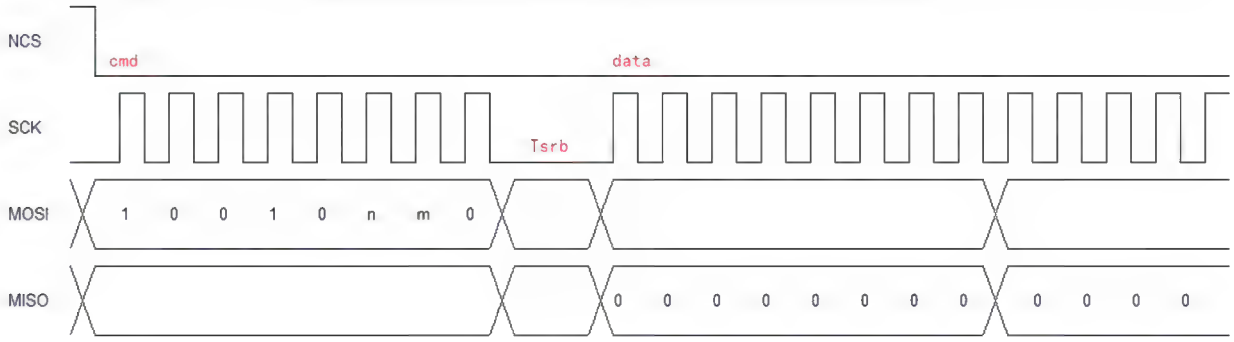
6.5 Read RX Buffer Instruction

The read RX buffer instruction provides a quick read of the receive buffer. The read RX buffer instruction actually has the following 4 possible values for determining the location of the address pointer. It is also necessary to clear the associated receive flag bit RXnIF to zero by itself after reading the associated register.

Table 6-1 Read the truth table of RX buffer instruction nm bit

Nm bit	Initial read address	Read number n
00	R8_RXB0SIDL	5
01	R8_RXB0D0	8
10	R8_RXB1SIDL	5
11	R8_RXB1D0	8

Figure 6-5 Read RX buffer instruction timing diagram



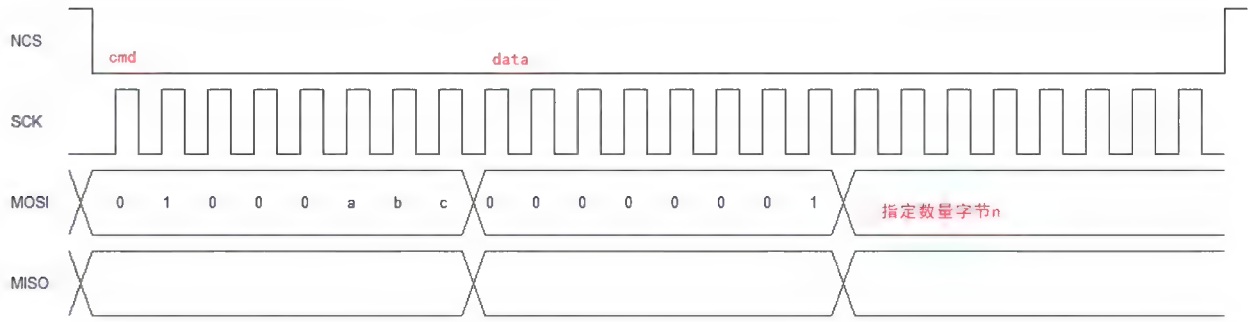
6.6 Load TX Buffer Instruction

The Load TX Buffer instruction speed sets the write address pointer to one of the following six addresses to quickly write data to the transmit buffer.

Table 6-2 Load the truth table of TX buffer instruction abc bit

Abc bit	Initial write address	Write number n
000	R8_TXB0SIDL	5
001	R8_TXB0D0	8
010	R8_TXB1SIDL	5
011	R8_TXB1D0	8
100	R8_TXB2SIDL	5
101	R8_TXB2D0	8

Figure 6-6 Load TX buffer instruction timing diagram



6.7 Request to Send (RTS) Instruction

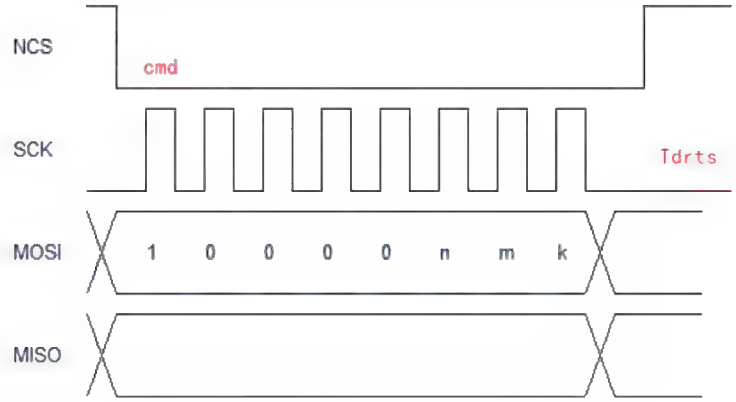
Use the RTS command to initiate message sending from one or more transmit buffers. The CH9431 is selected by setting the CS pin low, after which the RTS command byte is sent to it. The last three bits of the command show which transmit buffers are enabled for transmission.

This command sets the TXREQ bit of R8_TxBnCTRL corresponding to the buffer to 1. One or all three bits can be set to 1 with a single RTS command. the command is ignored if the last three bits of the RTS command are sent with a value of 000.

Table 6-3 Request to Send (RTS) instruction kmn bit truth table

Kmn bit	Execution result
K bit is 1	R8_TxB0CTRL TXREQ bit is 1
m bit is 1	R8_TxB1CTRL TXREQ bit is 1
n bit is 1	R8_TxB2CTRL TXREQ bit is 1

Figure 6-7 Request to Send (RTS) instruction timing diagram



6.8 Read Status Instruction

The read status instruction allows a single instruction to access commonly used message receive and transmit status bits.

The CH9431 is selected by setting the CS pin low, and then a read status command byte is sent to the CH9431. After the command byte is sent, wait for a specified time Tsc before reading it, and the CH9431 will return an 8-bit data containing status information.

Figure 6-8 Read status instruction timing diagram

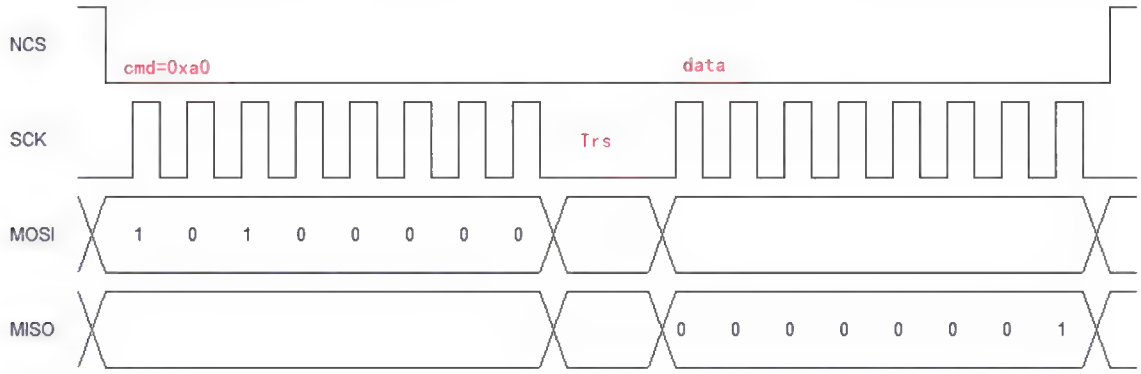


Table 6-4 Bits mapped in the byte returned by the read status instruction

Data bit in return byte	Mapping
0	RX0I bit of SYSINTF
1	RX1I bit of SYSINTF
2	TXREQ bit of R8_TXB0CTR
3	TX0I bit of SYSINTF
4	TXREQ bit of R8_TXB1CTRL
5	TX1I bit of SYSINTF
6	TXREQ bit of R8_TXB2CTRL
7	TX2I bit of SYSINTF

6.9 Read RX Status Instruction

The RX Status command is used to quickly determine the filter that matches the message and message type (standard, extended, or remote frame). After the command byte is sent, it needs to wait for the specified time Tsc to be read, and the CH9431 will return 8-bit data containing status information.

Figure 6-9 Read RX status instruction timing diagram

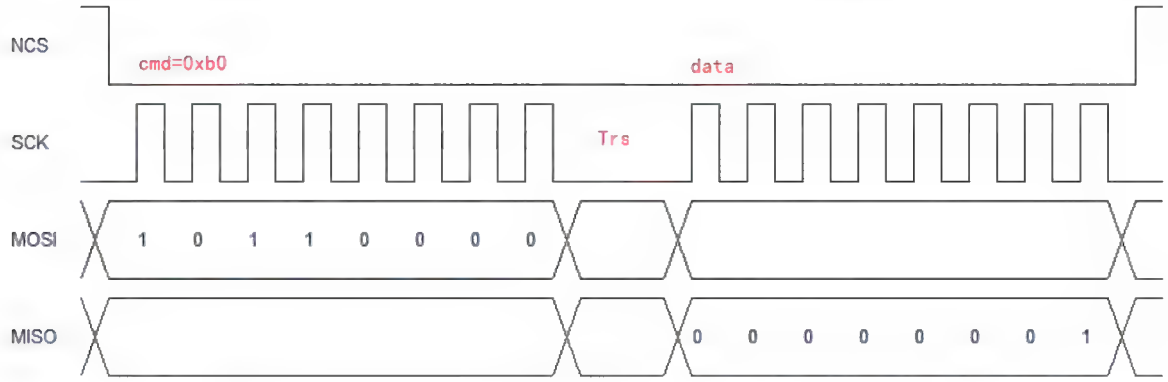


Table 6-5 Bits mapped in the byte returned by the read RX status instruction

Data bit in return byte	Mapping
0-2	Filter matching, see table 6-6.
3	The data frame is 0, the remote frame is 1.
4	The standard frame is 0, the extended frame is 1.
5	Reserve
6	RX0I bit of SYSINTF
7	RX1I bit of SYSINTF

Table 6-6 Filter matching truth table

0-2 bit	Filter matching
000	RXF0
001	RXF1
010	RXF2
011	RXF3
100	RXF4
101	RXF5
110	RXF0 (Rollover to RXB1)
111	RXF1 (Rollover to RXB1)

6.10 Bit Modification Instruction

The Bit Modify command sets or clears individual bits in specific status and control registers. The command is not valid for all registers.

The device is selected by setting the CS pin low, after which a bit modification command byte is sent to the CH9431. After the command byte is sent, the register address, mask byte, and data byte are sent in sequence. The mask byte determines which bit in the register is allowed to be modified. A '1' in the mask byte allows the corresponding bit in the register to be modified; a '0' prohibits modification.

The data byte determines the final result of register bit modification. If the corresponding bit of the mask byte is '1', a '1' in the data byte means that the corresponding bit of the register is set to 1, while a "0" will clear the bit.

The registers are not updated immediately after the operation of the bit modification instruction, and it is necessary to wait for the program check before updating the data to the specified registers. Therefore, it is not possible to write or modify a large number of registers in a short period of time.

Figure 6-10 Bit modification instruction timing diagram

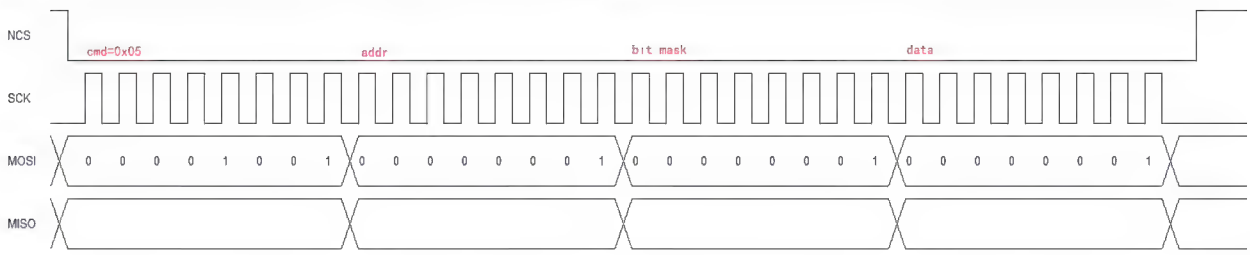


Table 6-7 SPI instruction set

Instruction name	Instruction format		Description
Reset	1100	0000	Reset the internal registers to their default state and set the device to configuration mode.
Read	0000	0011	Read data from the register at the specified address.
Read RX buffer	1001	0nm0	When reading the receive buffer, placing an address pointer at one of the 4 addresses indicated by "n, m" can reduce the overhead of general read commands.
Write	0000	0010	Writes data to the register at the beginning of the specified address.
Load TX buffer	0100	0abc	When loading the send buffer, placing an address pointer at one of the six addresses indicated by "a, b, c" can reduce the overhead of general write commands.
RTS (request to send message)	1000	0nmk	Instruct the control area to start sending the message sending sequence in any sending buffer. K is 1, requesting to send TXB0, M is 1, requesting to send TXB1, and N is 1, requesting to send TXB2.
Read state	1010	0000	Quick query command, which can read some status bits about sending and receiving functions.
RX state	1011	0000	Quick query command to determine the matching filter and the

			type of received message (standard frame, extended frame or remote frame).
Bit modification	0000	0101	Allow the user to set or clear individual bits in special registers. Note: This command is not applicable to all registers. Executing this command on registers that do not allow bit modification will be invalid. See the Register List for the applicable registers.

7. Operating Mode

The CH9431 has 7 operating modes, which are:

1. Normal operating mode;
2. Configuration mode;
3. Listen-only mode;
4. Loopback mode;
5. Shallow sleep mode;
6. Deep sleep mode;
7. Firmware update mode.

Selection of operating modes other than firmware update mode can be performed by operating the REOOP bit of R8_SYSCTRL (see the register detail table for details).

Immediately after changing the operating mode, it will determine whether it takes effect or not, and if it does, the unsent messages will not be sent again.

7.1 Normal Operating Mode

This mode is the standard operating mode of the CH9431. The device is in this mode to be able to transmit messages on the CAN bus. The normal operating mode can only be set in the configuration mode.

7.2 Configuration Mode

The CH9431 must be initialized before normal operation. The device can only be initialized in configuration mode. The device automatically enters configuration mode at power-up or reset.

The device can also be put into configuration mode from any mode other than sleep mode by setting the REQOP bit of R8_SYSCTRL to '001'.

The following registers can only be modified in configuration mode:

- R8_BTIMER1, R8_BTIMER2, R8_BTIMER3
- R8_TXRTSCTRL
- R8_RXFxSIDL
- R8_RXFxSIDH
- R8_RXFxEIDL
- R8_RXFxEIDH
- R8_RXMxSIDL
- R8_RXMxSIDH
- R8_RXMxEIDL
- R8_RXMxEIDH

7.3 Listen-only Mode

In this mode, the CAN controller can receive, not send messages externally, and is always in the hidden bit

externally, which avoids impacting the bus, but the messages can be received by the controller of the node where it is located. Usually the listen-only mode is used for CAN bus state analysis.

7.4 Loopback Mode

In this mode, the CAN controller can send messages externally and cannot receive external messages, but the sent messages can be received by the controller of the node where it is located, and the receive filtering mechanism is effective. Usually loopback mode is used for CAN controller send/receive test.

7.5 Shallow Sleep Mode

The shallow sleep mode allows you to listen to messages on the CAN bus and wake up after receiving them while keeping power consumption as low as possible. It is also possible to wake up by checking the chip select pin. After entering sleep, the pull-up resistor on the SI pin will be switched to a pull-down resistor, and at this time, the level on the SI bus will be read as low, and you can determine whether you have successfully entered sleep by reading the SI pin level. The previous operating mode is maintained after waking up. Shallow sleep mode can be entered only in normal operation mode and configuration mode. The CH9431 in sleep can be woken up by pulling down the CS pin, which should be pulled down for at least T_{CWK} time before pulling the CS pin high again. Wait for the SI pin to become a pull-up resistor again before continuing to issue commands via SPI.

7.6 Deep Sleep Mode

In deep sleep mode, the CAN controller does not support listening to the CAN bus and can only be woken up by checking the chip select pin. This mode has the lowest power consumption. After entering sleep, the pull-up resistor on the SI pin will be switched to a pull-down resistor, and the level on the SI bus is read as low at this time, and you can determine whether you have successfully entered sleep by reading the level on the SI pin. The previous operating mode is maintained after waking up. Deep sleep mode can be entered only in normal operation mode and configuration mode. The CH9431 in sleep can be woken up by pulling down the CS pin. The CS pin should be pulled down for at least T_{CWK} time and then pulled high again. Wait for the SI pin to become a pull-up resistor again before continuing to issue commands via SPI.

7.7 Firmware Update Mode

The user's main program should reserve an API interface to enter update mode, which can be found in the routine. The update mode cannot be entered by setting the REQOOP bit in R8_SYSCTRL, and the mode can only be entered by a fixed checksum step. After successful entry, the OPMOD bit in R8_SYSSTAT will be set to 110. update mode will update the firmware via the CAN bus.

8. Message Transmitting Function Description

The CH9431 has 3 transmit buffers that can be written to and queued for transmission and supports stopping transmission. It supports sending data frames, extended frames and remote frames.

8.1 Single-trigger Mode

The single trigger mode ensures that the message will only be sent once. Normally, if a CAN message fails arbitration or is corrupted by an error frame, the message will attempt to be sent again. Once single-trigger mode is enabled, the message will only be sent once, regardless of arbitration failure or error frame generation. Single-trigger mode is enabled by setting the OSM bit of R8_SYSCTRL.

8.2 Transmit Priority

Transmit Priority is the priority of the CH9431's internal wait for a transmit message. It is independent of the message arbitration priority inherent in the CAN protocol. TXB0 has the highest priority, TXB1 the next highest, and TXB2 the lowest priority.

8.3 Transmit Priority

At a minimum, the R8_TXBnSIDH, R8_TXBnSIDL, and R8_TXBnDLC registers must be loaded with data. If the message contains data bytes, the R8_TXBnDm register shall also be loaded. If the message uses an extended identifier, the R8_TXBnEIDm register should be loaded and the bit EXIDE of the R8_TXBnSIDH register should be set to 1.

Prior to the message being sent, the MCU should initialize the TXInE bit of R8_SYSINTE in order to enable or disable interrupt generation when the message is sent.

The TXREQ bit of R8_TXBnCTRL must be cleared to zero (indicating that there are no messages waiting to be sent in the transmit buffer) before writing to the transmit buffer.

8.4 Start Transmitting

By setting bit TXREQ of R8_TXBnCTRLD to 1, message transmitting can be initiated for the corresponding buffer.

When bit TXREQ of R8_TXBnCTRL is set to one, the ABTF bit of R8_TXBnCTRL, the MLOA bit of R8_TXBnCTRL, and the TXERR bit of R8_TXBnCTRL are automatically cleared.

Setting bit TXREQ of R8_TXBnCTRL to 1 will not initiate message transmission. It will only mark the message buffer as ready to send. A message transmission is initiated only when the device detects that the bus is idle.

After the message is successfully sent, bit TXREQ of R8_TXBnCTRL will be cleared to zero, bit TXnIF of R8_SYSINTF will be set to one, and an interrupt will be generated if bit TXnIE of R8_SYSINTE is set to one. If the message fails to be sent, bit TXREQ of R8_TXBnCTRL will remain set to 1, indicating that the message is still waiting to be sent. In this case, one of the following condition flags will be set to 1:

- If the message has started to be sent but an error occurs, bit TXERR of R8_TXBnCTRL and bit MERRF of R8_SYSINTF will be set to 1, and the device will generate an interrupt at the INT pin after bit MERRE of R8_SYSINTE is set to 1.
- If the bus arbitration of sending message fails, the bit MLOA of R8_TXBnCTRL will be set to 1.

8.5 TXnRTS Pin

TXnRTS is an input pin with a pull-up resistor that can be configured as follows:

- Request to Send Input Pin as an alternate method of initiating the sending of a message in either send buffer
- Standard Digital Input Pin (can be connected to a key)

8.6 Abort Transmitting

The MCU can request to abort the transmitting of messages in a particular message buffer by clearing the TXREQ bit of the corresponding R8_TXBnCTRL in that buffer to zero. It can also request to abort all messages waiting to be sent by setting bit ABAT of R8_SYSCTRL to one. The user must reset this bit to continue transmitting messages (usually after confirming that all TXREQ bits are cleared). The ABAT flag of R8_SYSCTRL is only set to 1 if the message is aborted by setting the ABAT bit of R8_SYSCTRL.

Only messages that have not yet started to be sent can be aborted. A message is aborted as soon as it is not sent successfully (e.g., arbitration failure or interruption due to an error frame).

9. Message Receiving Function Description

The CH9431 has 2 receive buffers. Each receive buffer is equipped with multiple receive filters. the RXB1 receive buffer supports another receive region configured as the RXB0 buffer.

9.1 Message Buffer

The CH9431 has a separate internal receive buffer in addition to the 2 independent receiving buffers RXB0 and RXB1 to additionally cache a packet that meets the filter conditions. The packet is automatically loaded into RXB0 or RXB1 after clearing the RX0IF bit or the RX1IF bit of R8_SYSINTF, and it will reset the RX0IF bit or the RX1IF bit of R8_SYSINTF, at which time the INT pin will continue to hold a valid level if the INT pin is valid due to the RX0IE bit or the RX1IE bit being enabled.

9.2 Reception Flag

When a message is transmitted to a receive buffer, the RXnIF bit of R8_SYSINTF corresponding to that receive buffer is set to 1. Once the message in the buffer has been processed, the MCU must clear this bit to receive the next message. Ensure that the CH9431 will not load a new message into the receive buffer until the MCU has finished processing the previous message.

If the RXnIE bit of R8_SYSINTE is set to 1, the device generates an interrupt on the INT pin indicating that the received message is valid. Also, if configured as a receive buffer full interrupt pin, the corresponding RXnBF pin is pulled low.

9.3 Rollover Function

The R8_RXB0CTRL register can be configured so that if another valid message is received after RXB0 already contains a valid message, the new message is rolled over to RXB1 unless RXB1 already has an unprocessed message.

9.4 Standard Frame and Extended Frame Filter Settings

Enable this filter by setting the EN bit of R8_RXFnSIDH to 1. If it is 0, the filter is disabled and will not receive messages.

Set whether this filter filters extended frames or standard frames by setting the EXIDE bit of R8_RXFnSIDH of the receive filter to 0 or 1. If 0, the filter will only receive standard frames if 1, the filter will only receive extended frames.

The filter for RXB0 consists of a receive mask register RXM0 and two receive filter registers RXF0 and RXF1.

The filter for RXB1 consists of one receive mask register, RXM1, and four receive filter registers, RXF2, RXF3, RXF4, and RXF5.

9.5 Reception Match

When the message is received, the R8_RXBnCTRL[3:0] bits will indicate the number of the receive filter that enabled this receive operation and whether or not the received message was remote transfer request.

10. Interrupt and Error Function Description

The CH9431 has 8 interrupt sources. R8_SYSINTE register contains the interrupt enable bits that enable each interrupt source. R8_SYSINTF register contains the interrupt flag bits for each interrupt source. When an interrupt occurs, the INT pin will be pulled low by the CH9431 and remain low until the MCU clears all enabled interrupts. Interrupts are cleared only after the condition that caused the corresponding interrupt has disappeared.

It is recommended to use the bit modification command instead of the normal write operation when performing a reset operation on the flag bits in the R8_SYSINTF register. This is to avoid inadvertent modification of the flag bits during the execution of the write command, which may result in interrupt loss.

10.1 Interrupt Code Bit

The value of the ICOD bit of R8_SYSSTAT reflects the source of the interrupt waiting to be processed. If more than one interrupt occurs at the same time, the INT pin will remain low until the MCU resets all interrupts. The ICOD bit of R8_SYSSTAT will reflect the highest priority interrupt code currently waiting to be processed. There is an internal priority between interrupt sources; the lower the ICOD code value, the higher its interrupt priority. Once the highest priority interrupt condition is cleared, the ICOD bit will reflect the code of the next highest priority interrupt waiting to be processed. The ICOD bit reflects only those interrupt sources whose R8_SYSINTE interrupt enable position is 1.

Table 10-1 ICOD true value table

ICOD[2:0]	Boolean expression
000	$\overline{ERR} \bullet \overline{WAK} \bullet \overline{TX0} \bullet \overline{TX1} \bullet \overline{TX2} \bullet \overline{RX0} \bullet \overline{RX1}$
001	ERR
010	$\overline{ERR} \bullet WAK$
011	$\overline{ERR} \bullet \overline{WAK} \bullet TX0$
100	$\overline{ERR} \bullet \overline{WAK} \bullet \overline{TX0} \bullet TX1$
101	$\overline{ERR} \bullet \overline{WAK} \bullet \overline{TX0} \bullet \overline{TX1} \bullet TX2$
110	$\overline{ERR} \bullet \overline{WAK} \bullet \overline{TX0} \bullet \overline{TX1} \bullet \overline{TX2} \bullet RX0$
111	$\overline{ERR} \bullet \overline{WAK} \bullet \overline{TX0} \bullet \overline{TX1} \bullet \overline{TX2} \bullet \overline{RX0} \bullet RX1$

10.2 Transmit Interrupt

When the transmit interrupt is enabled (the TXnIE bit of R8_SYSINTE is 1), if the relevant transmit buffer is empty and ready for new message loading, the device will generate an interrupt at the INT pin. The TXnIF bit of R8_SYSINTF will be set to 1 to indicate the interrupt source. Clear the interrupt by clearing this bit.

10.3 Receive Interrupt

With the receive interrupt enabled (R8_SYSINTE's RXnIE bit set to 1), the device generates an interrupt on the INT pin if the message is successfully received and loaded into the associated receive buffer. At the same time the RXnIF bit of R8_SYSINTF will be set to 1 to indicate the interrupt source. The interrupt is cleared by clearing this bit to zero. If new data continues to be received at this time, the interrupt and interrupt pins will remain active.

If the corresponding receive interrupt pin, RX0IP or RX1IP, is enabled, the receive interrupt pin must first be pulled high when the corresponding interrupt flag bit is cleared, and then pulled low again after new data is loaded into the receive buffer.

10.4 Message Error Interrupt

If an error occurs during message transmission and reception, the message error flag bit (MERRF bit of R8_SYSINTF) will be set to 1. At this time, the device will generate an interrupt at the INT pin if the corresponding MERRE bit of R8_SYSINTE is also set to 1.

10.5 Wakeup Interrupt

If the CH9431 is in hibernation mode and the bus activity wake-up interrupt is enabled (WAKIE bit of R8_SYSINTE is 1), an interrupt is generated on the INT pin when activity is detected on the CAN bus by setting bit WAKIF of R8_SYSINTF to 1. This interrupt causes the CH9431 to exit hibernation mode. The interrupt is cleared by clearing the WAKIF bit to zero.
The CH9431 wakes up in sleep mode and maintains its previous operating mode.

10.6 Error Interrupt

When an error interrupt is enabled (ERRIE bit of R8_SYSINTE is 1), the device will generate an interrupt at the INT pin if an overflow occurs or the transmitter/receiver error state changes. The Error Flag (EFLAG) register will indicate one of the following error interrupt conditions.

10.6.1 Receiver Overflow

When a valid message is received (the message meets the receiving conditions of the receiving filter), but the receiving buffer related to the filter can't load a new message, overflow will occur. The RXnOVR bit of the corresponding R8_EFLAG will be set to 1, indicating that an overflow has occurred. This bit must be cleared by MCU.

10.6.2 Receiver Warning

R8_REC reaches MCU warning limit value 96.

10.6.3 Transmitter Warning

R8_TEC reaches MCU warning limit value 96.

10.6.4 Receive Passive Error

R8_REC exceeds the passive error limit of 127 and the device enters the passive error state.

10.6.5 Transmit Passive Error

R8_TEC exceeds the passive error limit of 127 and the device enters the passive error state.

10.6.6 Bus Off

R8_TEC exceeds 255 and the device enters the bus-off state.

10.6.7 Transmit and Receive Error Register Description

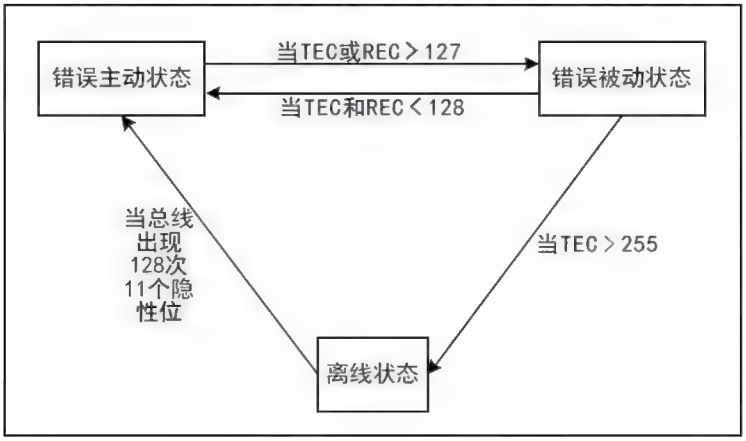
When R8_TEC and R8_REC of the Status Error Register are less than 128, the current CAN node is in the error active state and can participate in the bus communication normally and issue the active error flag when an error is detected.

When R8_TEC and R8_REC of the Status Error Register are greater than 127, the current CAN node is in an error passive state and is not allowed to issue an active error flag when an error is detected, but can only issue a passive error flag.

When R8_TEC of the Status Error Register is greater than 255, the current CAN node enters the offline state.

When the bus is monitored for 128 occurrences of 11 consecutive recessive bits, it returns to the error active state.

Figure 10-1 Transmit and receive error register description diagram



11. Clock and Bit Timing Description

11.1 Clock

CH9431 needs to be connected to an external 16MHz crystal and oscillator capacitor to provide a 16MHz input clock for the chip's internal clock oscillator. After internal frequency doubling and dividing, the final CAN bus operating reference frequency is 20MHz.

11.2 CAN Bus Timing Configuration

According to the CAN bus standard, each bit of time is divided into 4 segments: a synchronization segment, a propagation time segment, a phase buffer segment 1, and a phase buffer segment 2, respectively. these segments consist of the smallest time unit, Tq. The CAN controller monitors the CAN bus changes by sampling them, and synchronizes them by the edges of the frame starting bit.

CH9431 re-divides the above 4 segments into 3 segments, which are as follows:

Synchronization Segment (SS): That is, the synchronization segment in the CAN standard, which is fixed to 1 minimum time unit, and the bit jumps expected under normal circumstances occur within this time segment.

Time Segment 1 (TS1): Contains the propagation time segment and Phase Buffer Segment 1 of the CAN standard, which can be set to contain from 1 to 16 minimum time units, and can be automatically extended to compensate for the positive phase drift caused by the frequency accuracy errors of different nodes on the CAN bus. This time segment ends at the sample point position.

Time Segment 2 (TS2): This is the phase buffer segment 2 of the CAN standard, which can be set to contain 1 to 8 minimum time units and can be shortened automatically to compensate for the negative phase drift caused by the frequency accuracy errors of the different nodes on the CAN bus.

The resynchronization jump width (SJW), which is an upper limit on the number of minimum time units that can be extended and reduced in each bit, can be set in the range of 1 to 4 minimum time units.

All of the above parameters can be configured in the CAN bus timing register.

The final CAN bus baud rate calculation formula is:

20MHz / ((R8_BTIMER2.TS1 + 1 + R8_BTIMER3.TS2 + 1 + 1) * (R8_BTIMER1.BRP + 1))

12. Electrical Characteristics

12.1 Absolute Maximum Value

(Critical or exceeding the absolute maximum value will likely cause the chip to work improperly or even be damaged)

Name	Parameter	Min.	Max.	Unit
V_{DD}	Operating power supply voltage	-0.3	4.0	V
T_A	Ambient temperature during operation	-40	85	℃
T_S	Ambient temperature during storage	-40	125	℃
T_J	Junction temperature range	-40	105	℃
V_{ESD}	Electrostatic discharge voltage of ESD (mannequin, non-contact)	4K		V
I_{VDD}	Total current through VDD power supply line (supply current)		150	mA
I_{VSS}	Total current through VSS ground (outflow current)		150	mA
I_{IO}	Sink current on any I/O and control pins		25	mA
	Source current on any I/O and control pins		-25	mA

12.2 Electrical Parameters (Test condition: T_A=25℃, V_{DD}=3.3V)

Name	Parameter	Condition	Min.	Typ.	Max.	Unit
V_{DD}	Standard operating voltage	External CAN transceiver supports I/O levels at operating voltage V_{DD}	1.8	3.3	3.6	V
V_{IH}	Standard I/O pin, input high voltage.	V_{DD} = 3.3V	2.0	-	V_{DD}	V
		V_{DD} = 1.8V	1.3	-	V_{DD}	
	FT (tolerant 5V) I/O pin, input high level voltage	V_{DD} = 3.3V	1.9	-	5.0	V
		V_{DD} = 1.8V	1.2	-	5.0	
V_{IL}	Standard I/O pins, input low voltage	V_{DD} = 3.3V	0	-	1.0	V
		V_{DD} = 1.8V	0	-	0.6	
	FT (tolerant 5V) I/O pin, input low level voltage	V_{DD} = 3.3V	0	-	1.0	V
		V_{DD} = 1.8V	0	-	0.5	
R_{PU}	I/O pull-up equivalent resistance		30	40	50	kΩ
R_{PD}	Pull-down equivalent resistance		30	40	50	kΩ
C_{IO}	I/O pin capacitance		-	5	-	pF
I_{DD}	Operating current		-	6.7	-	mA
I_{DDS}	Shallow sleep current		-	4.6	-	mA
I_{DDDS}	Deep sleep current		-	4.5	-	uA
$V_{hys(RESET)}$	RESET Schmitt trigger voltage		150	-	-	mV

	hysteresis					
$T_{F(NRST)}$	RESET input can be filtered pulse width		-	-	100	ns
$T_{NF(NRST)}$	RESET input cannot filter pulse width		300	-	-	ns

12.3 Timing Parameters

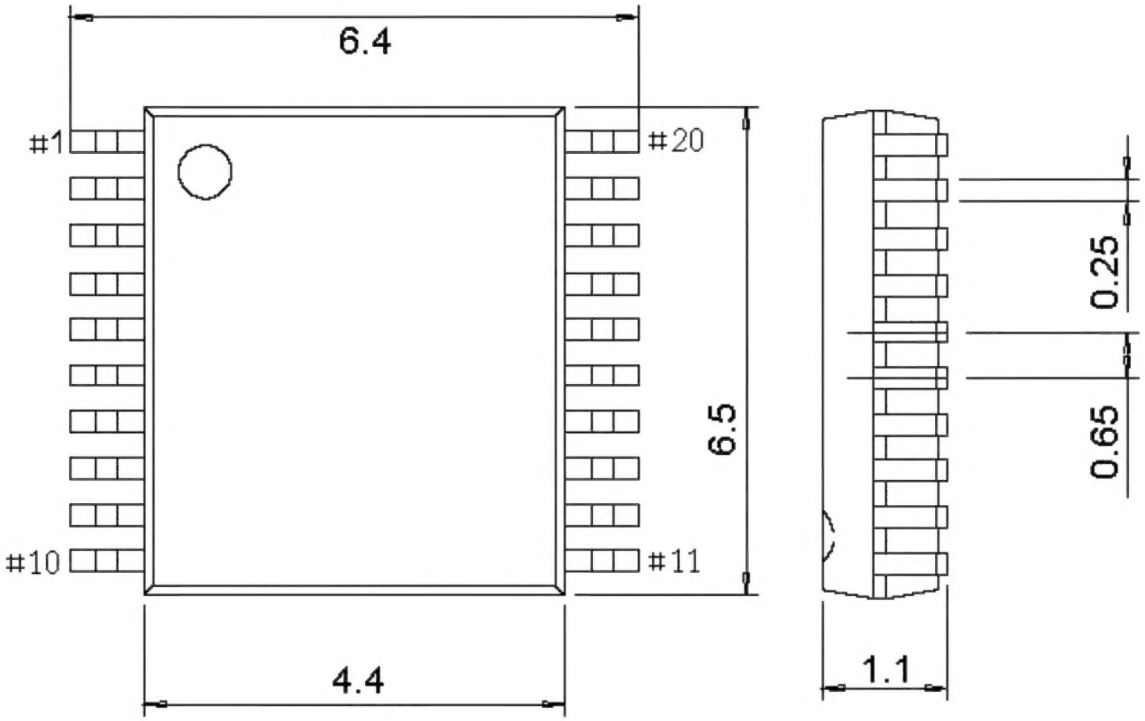
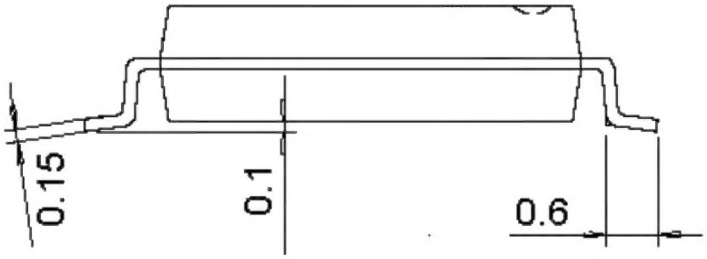
Name	Parameter	Min.	Typ.	Max.	Unit
T_{RD}	Reset time	13	16	38	ms
T_{WAK}	Wake-up time to exit from low-power state		2	3	ms
T_{CWK}	Duration of chip select low when exiting from low power state	300			ns
T_{SX}	The interval between the end of a command operation and the start of the next command	500			ns
T_{SC}	The interval between issuing address bytes and starting reading data bytes in a read command	150			ns
T_{SRB}	In the READ RX buffer command, the interval between issuing command bytes and starting to read data bytes.	500			ns
T_{RS}	The interval between the issuance of the read status or read RX status command and the start of data reading	1000			ns
T_{DRTS}	After RTS command is issued, interval time before next command	1500			ns

12.4 SPI Interface Parameters

Name	Parameter	Min.	Typ.	Max.	Unit
$1/t_{SCK}$	SPI clock frequency			12	MHz
$t_{SU(CS)}$	CS setup time	30			ns
$t_{h(CS)}$	CS holding time	30			ns
$t_{SU(SI)}$	Data input setup time	4			ns
$t_{h(SI)}$	Data input holding time	4			ns

13. Package Information

Note: All dimensions are in millimeters.

The pin center spacing values are nominal values, with no error. Other than that, the dimensional error is not greater than the greater of ±0.2mm.

13.1 TSSOP20